Are those California's Houses in 1990 located near the Ocean more Expensive, on Average?_Code

December 18, 2020

1 Project 1

1.1 Part 1

1.1.1 Summary

In this paper we analyze the California 1990 census data set. Our main goal is to find any relevant correlation between the price of a house the in california of 1990 and the its geographical location in reference to the pacific ocean. Our previous economic knowledge suggest that those house near the ocean tend to be more expensive, on average, but this may be the case that these properties are bigger or located near important economic centers (like downtown Los Angeles) and other factors that may move the price along the location. To purge the house price from these factors we add a set of control variables and later with the help of maps visualize where the most expensive house are concentrated and even if some of them are located near the ocean but in fact are cheap. Moreover, California is located over an important geological failure called the San Andreas fault zone, so we argue that a property investment near a zone where earthquakes are common drives the price down since there is an added risk when purchasing the house. Another map using the geological failures known up to 1990 may help to clarify this argument.

1.1.2 Introduction

This paper analyzes empirically data from California in order to test the hypothesis that those houses locates near the coast tend to be more expensive, on average. We suggest that those houses near the ocean tend to be more expensive, on average, but this may be the case that these properties are bigger or located near important economic centers (like downtown Los Angeles) and other factors that may move the price along the location.

To purge the house price from these factors we add a set of control variables and later with the help of maps visualize where the most expensive houses are concentrated. This tool helps to have a quick look of the distribution of houses in California, moreover we can appreciate an important finding: some of the houses that are located near the ocean are in fact cheap. We then suggest that these properties tend to be cheaper due to its location near to important geological failures. One of the most dangerous of them is called the San Andreas fault zone, so we argue that a property investment near a zone where earthquakes are common drives the price down since there is an added risk when purchasing the house. Another map using the geological failures (earthquakes) known up to 1990 may help to clarify this argument. Previous research have added valuable information to this field providing with valuables datasets and models that help to understand

better the determinants of the price of a house.

Zietz J., Emily Norman Zietz E., and Sirmans G. (2007) mention that house prices and the characteristics of a house share a close relationship. They also suggested that some characteristics of a house tend to be priced differently depending on the economic quantile of the potential buyer. The authors use OLS regression with quantile regressions and other types of econometric models (such as 2SLS), and also mention controls for spatial autocorrelation. This paper uses an extensive number of controls that may help to get better results for estimation purposes, differently we are using a significant smaller number of variables in order to test our hypothesis.

This paper narrows the current research in this field by considering the location to the ocean as a pottentially predictor of the price of a house. To test the hypothesis that those houses located near the ocean tend to be more expensive on average we use the California 1990 census that provides observations at the district level for the 58 counties in California; this dataset includes observations for the demographic level and characteristics of the houses at this district.The dataset used in this research can be downloaded from https://www.kaggle.com/camnugent/california-housing-prices/download.

[2]:
```
import pandas as pd
import numpy as np
import qeds
import geoplot as gplt
import mapclassify as mc
import seaborn as sns

import geoplot.crs as gcrs


import geopandas as gpd
import matplotlib.pyplot as plt

from shapely.geometry import Point

import folium
from folium.plugins import HeatMap

import requests
import pandas as pd
from bs4 import BeautifulSoup

from bokeh.io import output_notebook
from bokeh.plotting import figure, ColumnDataSource
from bokeh.io import output_notebook, show, output_file
from bokeh.plotting import figure
from bokeh.models import GeoJSONDataSource, LinearColorMapper, ColorBar,␣
 ↪HoverTool
from bokeh.palettes import brewer
output_notebook()
```

```
import json
import warnings
warnings.filterwarnings('ignore')


import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
# activate plot theme
import qeds
qeds.themes.mpl_style();
plotly_template = qeds.themes.plotly_template()
colors = qeds.themes.COLOR_CYCLE

# We will import all these here to ensure that they are loaded, but
# will usually re-import close to where they are used to make clear
# where the functions come from
from sklearn import (
    linear_model, metrics, pipeline, model_selection
)



import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import pandas as pd
import statsmodels.api as sm
from statsmodels.iolib.summary2 import summary_col
from linearmodels.iv import IV2SLS

from sklearn import linear_model
```

1.1.3 Reading the data

The dataset contains 20640 observations, with 10 variables. We focus our analysis on 12 variables to predict the price of a house (including 5 dummy variables). The variables longitude and latitude help to locate a given district. Also, note that ocean proximity is a categorical variable which may be difficult to work with, the best solution is to create dummy variables to evaluate the position of the houses to the ocean.

[3]:
```
#Import data as data frame from the working directory
#Change the data format to data frame
housing = pd.read_csv("housing.csv")
```

```
#To get a brief overview of the data
housing.head( )
```

[3]:

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	\
0	-122.23	37.88	41.0	880.0	129.0	
1	-122.22	37.86	21.0	7099.0	1106.0	
2	-122.24	37.85	52.0	1467.0	190.0	
3	-122.25	37.85	52.0	1274.0	235.0	
4	-122.25	37.85	52.0	1627.0	280.0	

	population	households	median_income	median_house_value	ocean_proximity
0	322.0	126.0	8.3252	452600.0	NEAR BAY
1	2401.0	1138.0	8.3014	358500.0	NEAR BAY
2	496.0	177.0	7.2574	352100.0	NEAR BAY
3	558.0	219.0	5.6431	341300.0	NEAR BAY
4	565.0	259.0	3.8462	342200.0	NEAR BAY

1.1.4 Cleaning the data

Looking for missing values we found that the number of total bedrooms column has some missing data. Since this dataset is from a census we may think that some of the missing data come from districts such as the industrial or the financial district where the majority of the infrastucture is working space such like warehouse or offices. Then, the best procedure in this situation is to remove the missing values since we want to predict the prices of the residential houses. Also, the introduction of dummy variables in place of the categorical data ocean_proximity is done in this part.

[4]:
```
#To drop any missing values, we do not want to use them for our analysis

hc = housing.dropna()
```

[5]:
```
#Express categorical variable ocean proximity as a set of dummy to include them␣
 ↪in our regression analysis
dummies = pd.get_dummies(hc.ocean_proximity)

hc[dummies.columns] = dummies
```

1.1.5 Choosing a Variable that can better explain the outcome.

For this part we are interested in chosing one of the variables that better explain the prices of the houses in California in 1990. We are going to focus in the median income of the households since it is obvious to think that the people who earn more money are the ones with a higher purchasing power then the ones that can purchase more expensive houses, the inverse usually is true. Before choosing an especific variable we want to have a brief look to the whole summary of the data.

1.1.6 Transforming Variables

To reduce skewness of the data, the mullticollinearity of variables and the evolution of the price with respect to the age of a house we conduct the following variables transformations.

[7]:
```
hc['ln_(house_value)'] = np.log(hc['median_house_value'])
hc['ln_(income)'] = np.log(hc['median_income'])
hc['housing_median_age^2'] = hc['housing_median_age'].pow(2)
hc['bedroom_rate'] = (hc['total_bedrooms'] / hc['total_rooms']) * 100
hc['household_rate'] = (hc['households'] / hc['population']) * 100
```

1.1.7 Summary Statistics

Table 1 shows the summary for the 20,433 observations in this sample, we can appreciate that the median house value oscillates between $\$14,999$ and $\$500,001$ with an average of $\$206,864.41$. This variable is widely dispersed, note that one standard deviation ($\$115,435.67$) is equivalent to half of the mean. The set dummies includes 9,034 houses located at less than one hectarea of the ocean, the mean price of houses in this area is $\$240,268$. Inland records those houses that located more in the center of California, it contains 6496 observations houses locate inland average a value of $\$124,896$. Near bay records 2270 houses near the bay of San Francisco, the average price in this area is $\$238,815$. Similarly, 2628 houses are located near the ocean, the average price here is $\$249,042$. The most expensive houses are located on islands, only 5 observations with average price of $\$380,440$. Graph 1 and graph 2 resume the finding of the dummies variables so far. The average median income reported for the households in this sample is $\$3.87$, but with a standard deviation of half the mean ($\$1.89$). For this variable the minimun value occurs at $\$0.49$) this means that the data is negatively skewed. Houses in the sample averaged an age of 28 years with a standard deviation of 12 years, once again almost half the mean. There are new houses and houses as old as 52 years, very common in the real state. The average total rooms recorded 2636 rooms for an average number of households of 499 and and average population of 1424 people.

[9]:
```
#Summary Stistics for all the variables
np.round(hc.loc[: , ['median_house_value', 'ln_(house_value)',
                                        'median_income',
                                        'ln_(income)',
                                        'housing_median_age',
                                        'housing_median_age^2',
                                        'total_rooms',
                                        'total_bedrooms',
                                        'bedroom_rate',
                                        'population',
                                        'households',
                                        'household_rate',
                    ]].describe(percentiles=None), 2).T[['count','mean', 'std',␣
 ↪'min', 'max']]
```

[9]:

	count	mean	std	min	max
median_house_value	20433.0	206864.41	115435.67	14999.00	500001.00
ln_(house_value)	20433.0	12.08	0.57	9.62	13.12

```
median_income          20433.0       3.87       1.90       0.50      15.00
ln_(income)            20433.0       1.24       0.47      -0.69       2.71
housing_median_age     20433.0      28.63      12.59       1.00      52.00
housing_median_age^2   20433.0     978.40     751.12       1.00    2704.00
total_rooms            20433.0    2636.50    2185.27       2.00   39320.00
total_bedrooms         20433.0     537.87     421.39       1.00    6445.00
bedroom_rate           20433.0      21.30       5.80      10.00     100.00
population             20433.0    1424.95    1133.21       3.00   35682.00
households             20433.0     499.43     382.30       1.00    6082.00
household_rate         20433.0      36.43       9.34       0.08     144.44
```

1.1.8 Quick Glance at Dummies

[10]:
```
o = [2270, 9034, 6496, 2628,5]
hc.iloc[:, 10:16].sum( )
```

[10]:
```
<1H OCEAN              9034.000000
INLAND                 6496.000000
ISLAND                    5.000000
NEAR BAY               2270.000000
NEAR OCEAN             2628.000000
ln_(house_value)    246929.992662
dtype: float64
```

---

2 Project 2

2.1 The message

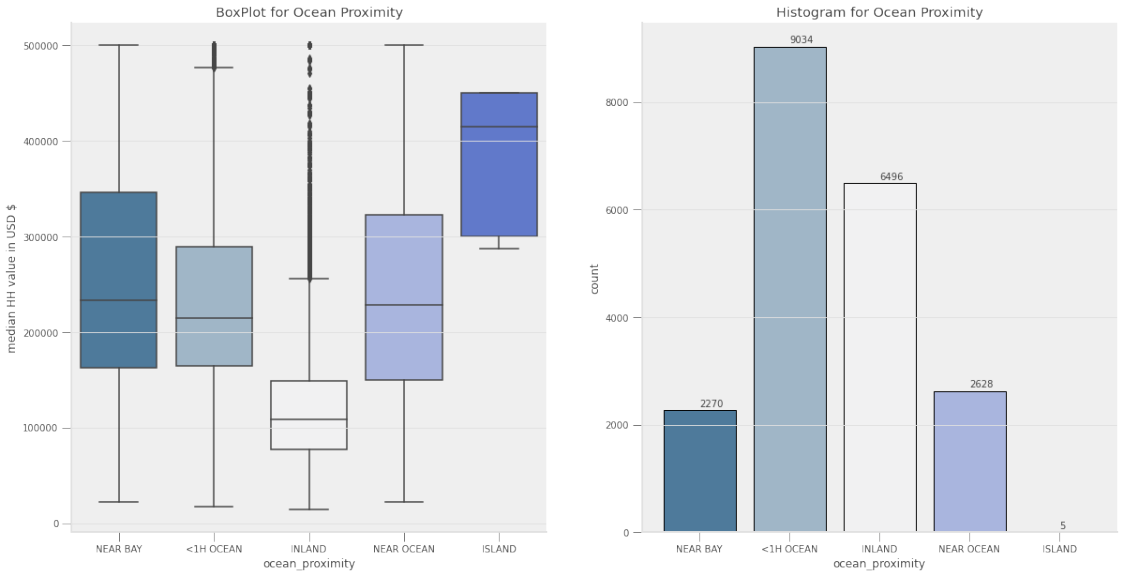
Our main goal is to estimate the price of a house in California in 1990 using a set of covariates. Then, our interest is on recognize any linear relationship between the independent variables, x, and each of the independent variables with the dependent variable,y. If there is any correlation between each of the x variables with the y variable then we may use these xs to estimate our y. Additionally, in the real world those houses near the ocean tend to have higher prices for obvious reasons. The dataset includes a variable that records the location of the house in reference to the ocean, we may use these categories as dichotomous in order to analyze how the price of a house varies accordingly to the position of the house to the ocean. In order to investigate these two situations we use a set of plots.

Boxplot and Histogram for Ocean Proximity

[11]:
```
fig, ax = plt.subplots(nrows=1, ncols=2, figsize=(20, 10))

sns.boxplot(data = hc, x ='ocean_proximity', y ='median_house_value', palette␣
↪=sns.diverging_palette(240, 260, n=5), ax = ax[0]).set( ylabel = 'median HH␣
↪value in USD $')
```

```
ax[0].set_title("BoxPlot for Ocean Proximity")
sns.countplot(hc.ocean_proximity, ax = ax[1], palette = sns.
 ↪diverging_palette(240, 260, n=5), edgecolor = 'black')
ax[1].set_title("Histogram for Ocean Proximity")

o = [2270, 9034, 6496, 2628,5]
for index, value in enumerate(o):
    plt.text( index, value + 70, s = str(value))

plt.plot( )
```

[11]: []



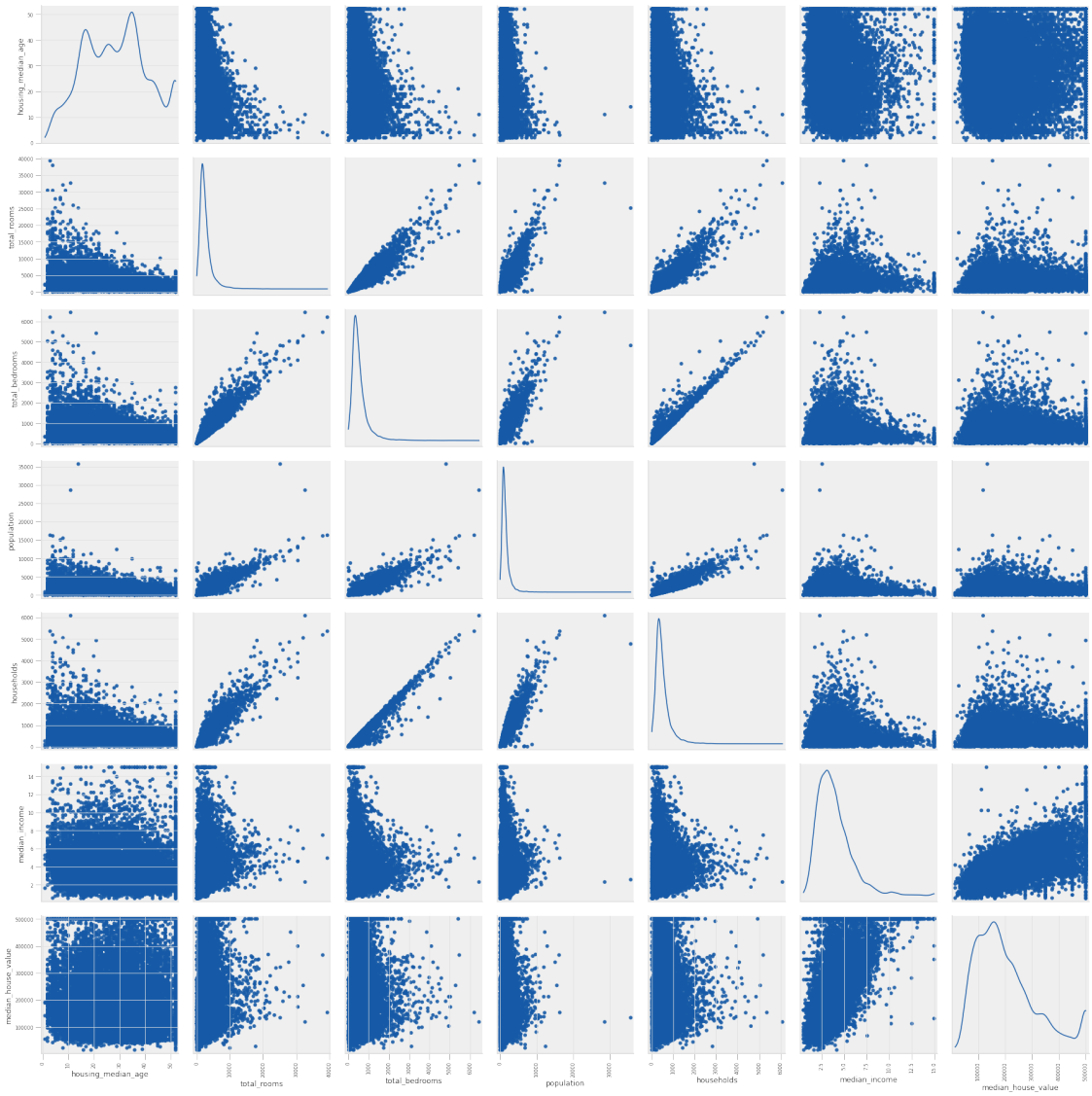
To appreciate the distribution of the data we plot the dummy variables for ocean proximity in two side by side graphs. The decision for a boxplot is to being able to have a quick overview of the data that is check for min and max values, median values and outliers. I complemented this boxplot by the sum of each dummy which yields the number of houses by position. Interestingly, most of the houses are less that one hectare of distance from the ocean and only 5 in island which turns to be the most expensive houses in the dataset with median value of around $\$415,000$. These graphs help to conclude with certainty that as the houses are closer to the ocean the value of the house tends to increase (compare inland to other variables). Then, including a set of dummy variables for location of the house relative to the ocean might help to get a better estimation of the house price while doing a regression analysis.

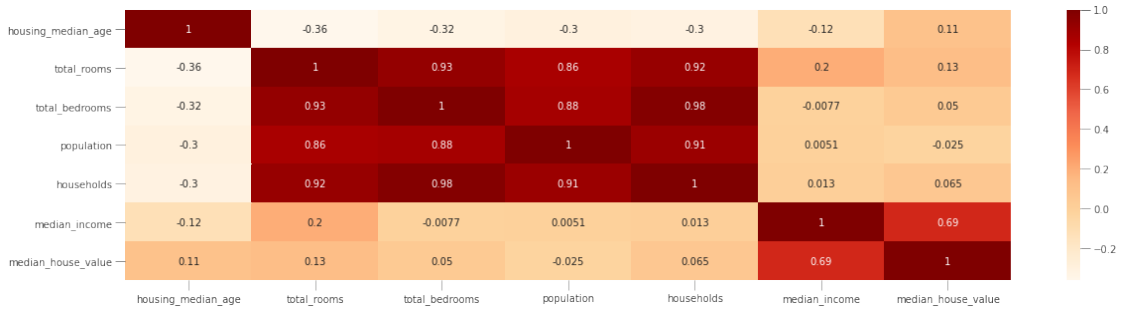
Scatter Plots and Correlation Matrix for the Independent Variables vs. Dependent Variable

[9]:
```
#Looking for perfect multicollinearity in the data
```

```
#Note that we may not include the set of dummy variables here since there are␣
 ↪categorical variables
#TWO PLOTS IN ONE OUTPUT ONE EXPLANATION FOR BOTH BELOW
pd.plotting.scatter_matrix( hc.iloc[: , 2:9], alpha = 0.9, diagonal = 'kde' ,␣
 ↪marker = "o", grid = True,  figsize = ( 25, 25 ) )
plt.tight_layout( )
plt.figure( figsize = (20,5) )
sns.heatmap( hc.iloc[: , 2:9].corr( ), annot=True, cmap = 'OrRd')
```

[9]: <AxesSubplot:>

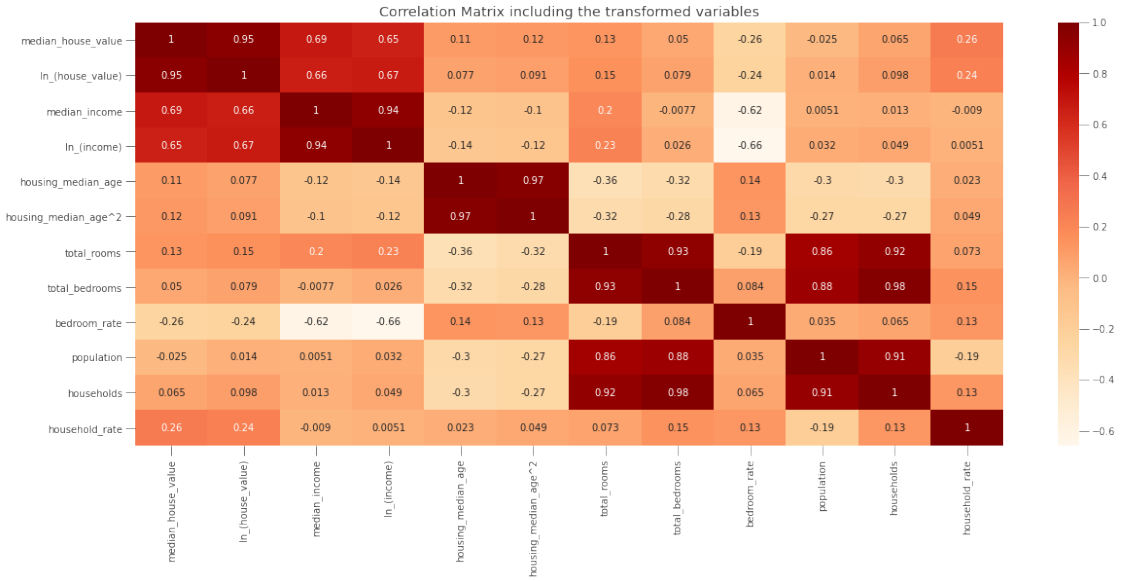




Additionally to the histogram and boxplot for ocean proximity, we would like to see if there is any correlation or linear relationship between the set of variables. To do this we plot a scatter plot matrix of the covariates, Xs, and the median house value, Y, against each other. Here, we noted that there are many covariates that hold linear relationships this is the case of total bedrooms and total rooms, this is obvious if we have more bedrooms we will increase the number of rooms. If we would conduct a strict regression analysis we may need to check for perfect collinearity, this drives us to the second plot where we can see that there is collinearity between some variables such as households and total rooms but not perfect collinearity between them. In the first graph the last column is our Y variable so we can use this columns to see the linear relationship between each x and the Y. Finally, we plot a kernel density in the diagonal just to see how the x is distributred and if there is any skewness or outlier, turns out the majority of the data is right skewed. Note the high multicollinearity between some variables, adopt transformations and compare the latter correlation matrix with a new correlation matrix.

[30]:
```
fig, ax = plt.subplots(figsize=(20, 8))
variables = hc.loc[: , ['median_house_value', 'ln_(house_value)',
                                        'median_income',
                                        'ln_(income)',
                                        'housing_median_age',
                                        'housing_median_age^2',
                                        'total_rooms',
                                        'total_bedrooms',
                                        'bedroom_rate',
                                        'population',
                                        'households',
                                        'household_rate',
                                        ]]
sns.heatmap( variables.corr( ), annot=True, cmap = 'OrRd', ax = ax)
ax.set_title("Correlation Matrix including the transformed variables")
```

[30]:
```
Text(0.5, 1.0, 'Correlation Matrix including the transformed variables')
```



Note how the variables that before shared high correlations now appear to be slightly correlated. There are some variables that share high correlations, avoid to mix them up when running OLS regressions in order to avoid biased coefficients.

2.2 Maps of California

For this part I would like to know how the districts are distributed in the map of California state. To do this I use a shape file from the U. S. census page that contains the counties around the country. I use the California state id which is 06 to plot the map of California with each county so this is the first layer. As a note I did not include the shape of the state since the counties are well defined in the geometry section that the border of California state is well defined as well.

Convert coordinates to Point Shape Objects

[31]:
```
hc["Coordinates"] = list ( zip (hc.longitude, hc.latitude ) ) #Transform the lat␣
 ↪and log into a tuple like coordinates
hc["Coordinates"] = hc["Coordinates"] .apply(Point) #Turn the tuple into a␣
 ↪Shapely point object
ghc = gpd.GeoDataFrame(hc, geometry = "Coordinates" ) #Convert the DataFrame␣
 ↪into a GeoDataFrame by calling the geopandas.DataFrame method
ghc.head( 2 ).round(2)
```

[31]:

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	\
0	-122.23	37.88	41.0	880.0	129.0	
1	-122.22	37.86	21.0	7099.0	1106.0	

	population	households	median_income	median_house_value	ocean_proximity	\
0	322.0	126.0	8.33	452600.0	NEAR BAY	

```
1      2401.0      1138.0           8.30            358500.0        NEAR BAY

   ...  ISLAND  NEAR BAY  NEAR OCEAN  ln_(house_value)  ln_(income)  \
0  ...       0         1           0             13.02         2.12
1  ...       0         1           0             12.79         2.12

   housing_median_age^2  bedroom_rate  household_rate  const  \
0                1681.0         14.66           39.13      1
1                 441.0         15.58           47.40      1

                   Coordinates
0  POINT (-122.23000 37.88000)
1  POINT (-122.22000 37.86000)

[2 rows x 22 columns]
```

For the last step note the importance of transform the coordinates to points this is valuable to plot these as points in the map.

Reading the First layer, USA counties Shapefile

[32]:
```
#County dataset from USA census gov page
#states contains counties california contains 58 counties
#THIS SHAPE FILE IS GOOD FOR COUNTIES

usa_cty = gpd.read_file("https://www2.census.gov/geo/tiger/GENZ2019/shp/
 ↪cb_2019_us_county_5m.zip")
usa_cty.head( 2)
```

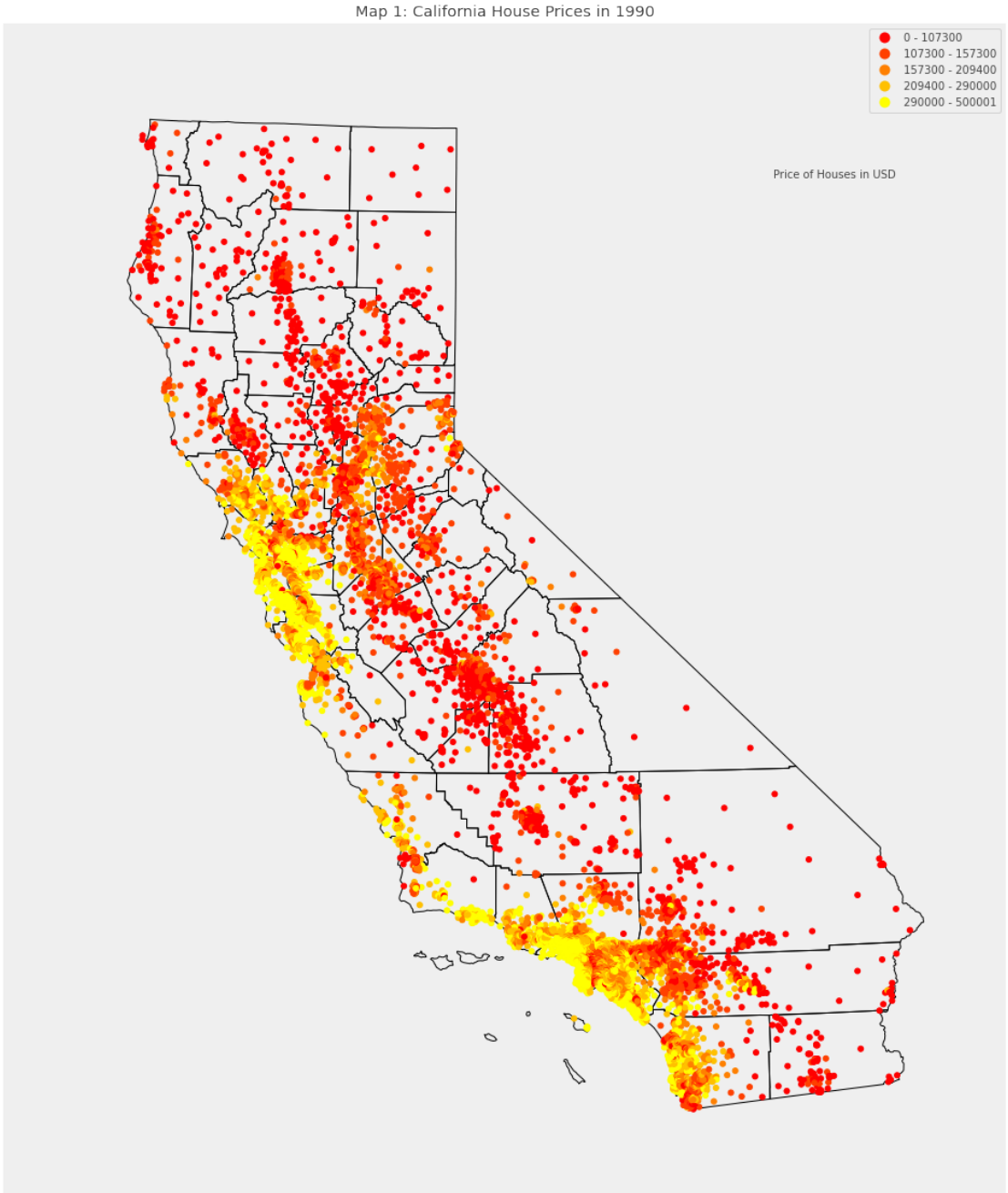
[32]:
```
  STATEFP COUNTYFP  COUNTYNS        AFFGEOID  GEOID       NAME LSAD  \
0      24      510  01702381  0500000US24510  24510  Baltimore   25
1      31      169  00835906  0500000US31169  31169     Thayer   06

        ALAND    AWATER                                           geometry
0   209650970  28758714  POLYGON ((-76.71131 39.37193, -76.62619 39.372...
1  1486153245   3025339  POLYGON ((-97.82082 40.35054, -97.36869 40.350...
```

Plotting the Maps

[33]:
```
#State of California has FIPS = 06
usa_cty= usa_cty.query("STATEFP == '06' ")

#Plot the Polyplot Map
ax = gplt.polyplot(usa_cty, projection = gcrs.AlbersEqualArea( ), figsize=(20,␣
 ↪20) )
scheme = mc.Quantiles( ghc ['median_house_value'], k=5)
gplt.pointplot( ghc,
               ax = ax,
```

```
                projection = gcrs.AlbersEqualArea( ),
                hue = 'median_house_value',
                scheme = scheme, cmap='autumn',
                legend = True
               )
ax.annotate( 'Price of Houses in USD', xy = ( 0.76, 0.85 ), xycoords = 'figure␣
 ↪fraction' )
ax.set_title( 'Map 1: California House Prices in 1990' )
```

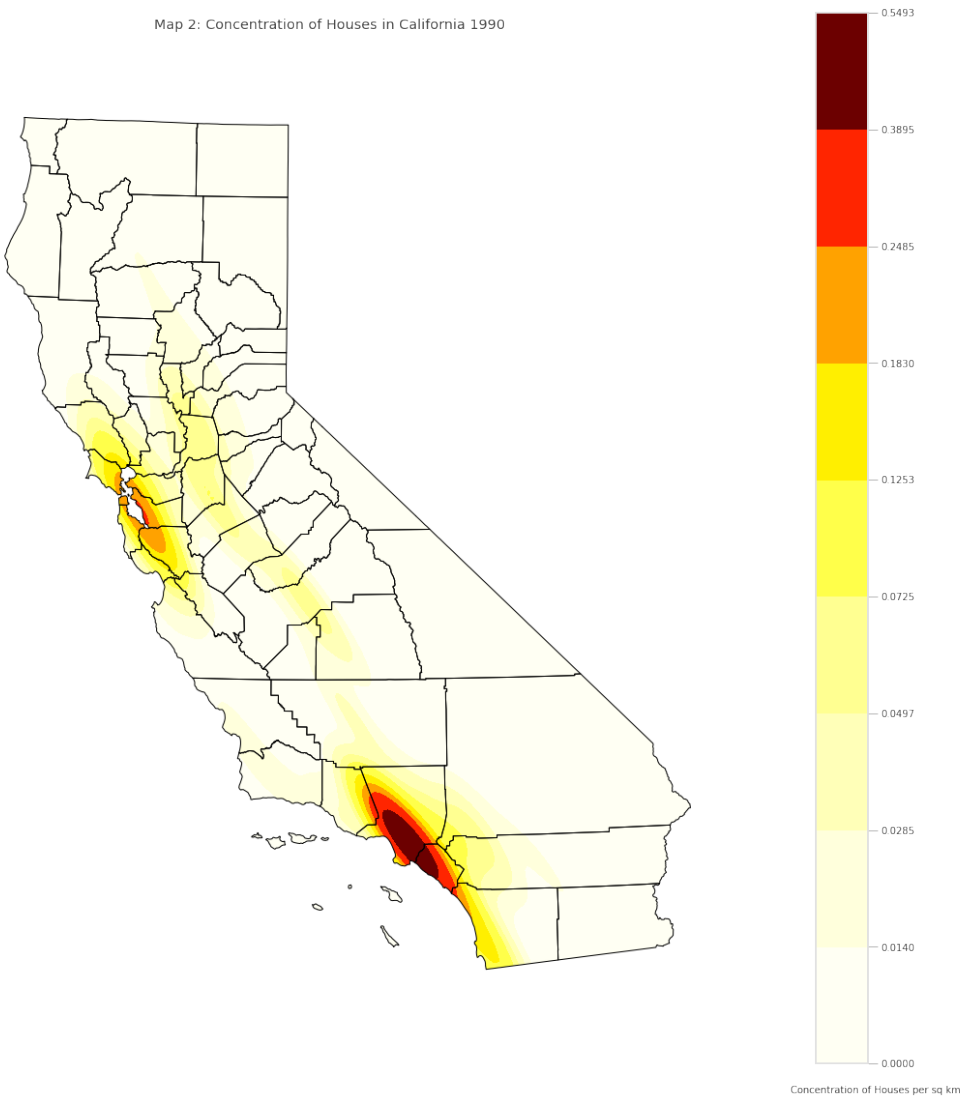
[33]: Text(0.5, 1.0, 'Map 1: California House Prices in 1990')



The map 1 shows the location of the houses in the Map of Califonia sorted by median house value. We sorted from the highest value to the lowest by quantiles, then we can appreciate that the houses in the 5th quantile are the expensive ones and are represented as yellow dots in the map 1. As suspected, the expensive houses are located near the ocean, however we can see that this is not exclusive since there are some dots from the 4th quantile near the ocean and even a few from the 1st quantile. This is very interesting since we can deduce that even location near the ocean is a good predictor for the price of the house this is not the only factor that drives the price. While this graph is very informative we have two concentrations the first one near San Francisco and the second one near Los Angeles, so it may be helpful to see which location concentrates most of the houses, since it may be the case that the location with more houses turns to be the one with more population. This takes us to map 2 and map 3 which is optional for HTML feature.

[34]:
```
ax = gplt.polyplot( usa_cty, projection = gcrs.AlbersEqualArea ( ), figsize = (␣
↪20,20 ), zorder = 4 )

gplt.kdeplot ( ghc, cmap = 'hot_r', ax = ax, shade = True, shade_lowest = True,␣
↪cbar = True, clip = usa_cty.geometry )

ax.annotate( 'Concentration of Houses per sq km', xy = ( 0.80, 0.06 ), xycoords␣
↪= 'figure fraction' )
ax.set_title("Map 2: Concentration of Houses in California 1990")
```

[34]:
```
Text(0.5, 1.0, 'Map 2: Concentration of Houses in California 1990')
```



This map uses a kernel density to show us which of the two locations mentioned before is the one that has more concentration of houses. Both locations are pretty packed with houses as we can see there are remarkable differences of these two location with the rest of the map, however the location near Los Angeles is the one with more observations. Los Angeles in the same manner as San Francisco host most of the expensive houses in the dataset, however the concentration of these houses is more near the ocean that the bay in San Francisco so those houses near the ocean may not be only be expensive for being close the ocean but for being in a more concentrated location for houses. This finding is only posible by loking at the two maps, map 1 and map 2 together.

---

3 Project 3

3.1 Part 2

3.1.1 Adding more information to my dataset from webscrapping.

Our main goal is to construct a regression equation that predicts the price of a house using a set of covariates that are relevant to this price and use this regression to make inferences on the coefficients on the set of dummies. I would like to sharpen this prediction by adding information about the crime rates per district in California. I expect to relate the crimes in the district to the price of a house. Then, for those districts with a higher crime rate I expect to see a lower house price, similarly more expensive houses may be located in or near safer areas. I would like to work with a website that provides information about the crime rates in California per distric in the year 1990, however this is difficult to obtain since the best that we can do is looking for information at the county level in 1990. In order to analyze the crime rates in 1990 we can extract information from the Open Justice website this is an open and freely resource from the California Department of Justice that provides information of the crimes in California since 1985, by county. Note that this website is designed to provide with the totals when selecting more than one county, then in order to extract information from one county the process must be done individually for each. Since our dataset is at the district level there are some locations that are in the border of the counties then it is difficult to merge this data with our dataset, the main difficulty is the location of our observations that are at an specific lat-lon level while the data from the website is at the county level. (url: https://openjustice.doj.ca.gov/exploration/crime-statistics/crimes-clearances)

3.1.2 Challenges of using this information and Reasons for not using it.

I do not need to run the program over time since I am working with data from 1990 so the data can be obtained by individual searching by county. The website can be scrapped but it is time consuming since has to be done individually for the 58 counties in California. We may write a code that looks in the webpage for each county and then analyze the new obtained data to see if those counties with more crime are near or far from the most expensive houses in California in 1990.

3.1.3 Web Scrapping from the given website.

We are going to use a website that provides information about earthquakes that took place in California from 1969 to 2007. The relvant years of information that we are going to use are in the range of 1969 to 1990. Note that we can use this information to sharpen our prediction of the the price of a House, that is, a house that is located over a geological failure or a well-known area of earthquakes may be cheaper that house located far away of this area. For the nature of the data the best approach is to use a map to check for this assumption.

Step 1: Request URL First we use the request library and call the website the response is received in HTML format but it is not easy to read we can check this by priting the content using print(response.content [: 2000]) for the sake of space I omit this last part.

[35]:
```
#Step 1: Request URL
web_url = 'http://wiki.stat.ucla.edu/socr/index.php/SOCR_Data_021708_Earthquakes'
response1 = requests . get ( web_url )
```

Step 2: More readable, soup object The content from the website is not that easy to read, so we can make it more readable by using the BeatifulSoup( ) method, the final result is the HTML content well structured.

[36]:
```
#Step 2: more readable, soup object
soup_object1 = BeautifulSoup (  response1.content )
```

Step 3: Extract information by elevant tags For this step we have to know how the website is structured in HTML format and the relevant tags that we are interested in, to do this we can inspect the element in the website, for our case the format is a table and the class is a wikitable. Note that the wikitable for our interest starts at index 1 not 0.

[37]:
```
#Step 3: extract by elevant tags
data_table1 = soup_object1 . find_all ( 'table' , 'wikitable') [1]
```

Step 4: Find all values by rows For this step we can separate the information by rows, using the relevant tag

. Further inspection sharpens our dataset to the rows 1 to 106 which is the last observation for the year 1990.

[38]:
```
#Step 4: Find all values by rows
all_values1 = data_table1.find_all('tr')
all_values1[:3]
```

[38]:
```
[<tr>
 <th> Date_(YYYY/MM/DD) </th><th> Time </th><th> Latitude </th><th> Longitude
</th><th> Depth </th><th> Mag </th><th> Magt </th><th> Nst </th><th> Gap
</th><th> Clo </th><th> RMS </th><th> SRC </th><th> EventID
 </th></tr>,
 <tr>
 <td> 1969/10/02 </td><td> 04:56:45.30 </td><td> 38.4978 </td><td> -122.6640
</td><td> 0.22 </td><td> 5.60 </td><td> ML </td><td> 38 </td><td> 104 </td><td>
52 </td><td> 0.22 </td><td> NCSN </td><td> -1003132
 </td></tr>,
 <tr>
 <td> 1969/10/02 </td><td> 06:19:56.39 </td><td> 38.4500 </td><td> -122.7535
</td><td> 5.14 </td><td> 5.70 </td><td> ML </td><td> 53 </td><td> 139 </td><td>
58 </td><td> 0.22 </td><td> NCSN </td><td> -1003135
 </td></tr>]
```

Step 5: Extract and save the required dataset by for loop The data that we are interest in can be stored in a dataframe. To do so we create an empty dataframe run an for loop and store the relevant information in the corresponding columns.

[39]:
```
#Step 5: extract and save the required dataset by for loop
```

```
Earthquakes = pd.DataFrame ( columns = [ 'Latitude', 'Longitude', 'Magnitude'] )␣
 ↪ #creates a container
ix = 0
#Last obs for 1990 is at index 106

for row in all_values1 [1:106] :
    values = row.find_all( 'td' )

    Latitude = float ( values[2]. text )
    Longitude = float ( values[3] .text )
    Magnitude = float ( values[5] .text )


    Earthquakes.loc[ix] = [ Latitude, Longitude, Magnitude ]

    #increase the index by 1
    ix += 1

Earthquakes.head( )
```

[39]:

	Latitude	Longitude	Magnitude
0	38.4978	-122.6640	5.6
1	38.4500	-122.7535	5.7
2	36.5903	-121.1905	5.1
3	36.9202	-121.4673	5.2
4	40.5415	-124.2763	5.3

3.1.4 - 3.15 Merging the scapped data and Visualization

For the nature of our original dataset we cannot simply merge these two data sets, then we use a maps to check for any correlation between the price and the concentration of earthquakes in California.

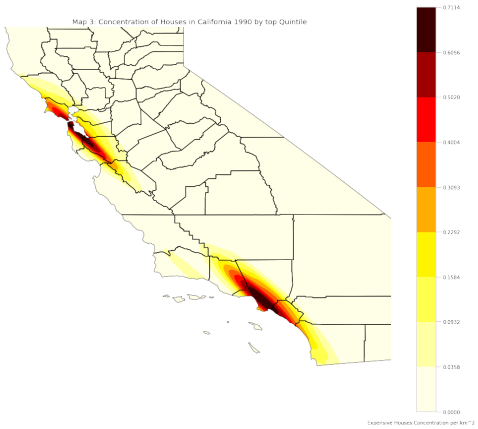
[40]:
```
#Plotting the Earthquake Map

eq = Earthquakes
eq["Coordinates"] = list ( zip (eq.Longitude, eq.Latitude ) ) #Transform the lat␣
 ↪and log into a tuple like coordinates
eq["Coordinates"] = eq["Coordinates"] .apply(Point) #Turn the tuple into a␣
 ↪Shapely point object
geq = gpd.GeoDataFrame(eq , geometry = "Coordinates" ) #Convert the DataFrame␣
 ↪into a GeoDataFrame by calling the geopandas.DataFrame method
geq.head(5)
```

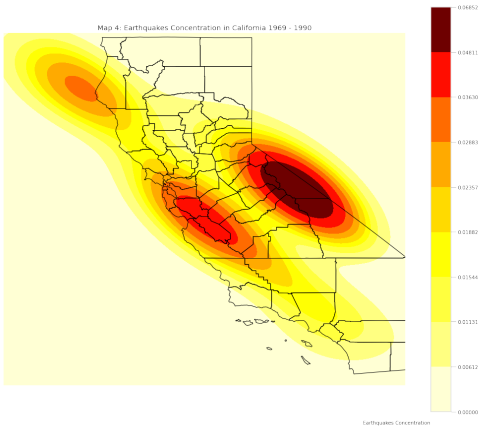
[40]:

	Latitude	Longitude	Magnitude	Coordinates
0	38.4978	-122.6640	5.6	POINT (-122.66400 38.49780)
1	38.4500	-122.7535	5.7	POINT (-122.75350 38.45000)

```
2   36.5903  -121.1905        5.1  POINT (-121.19050 36.59030)
3   36.9202  -121.4673        5.2  POINT (-121.46730 36.92020)
4   40.5415  -124.2763        5.3  POINT (-124.27630 40.54150)
```

[41]:
```
#Plotting the Maps

fig, ax = plt.subplots(1, 2, figsize = (40, 15) )

#First HeatMap: Concentration of the most Expensive Houses

ax[0] = gplt.polyplot( usa_cty, zorder = 1, ax = ax[0] )
gplt.kdeplot ( ghc[ghc[ 'median_house_value'] >= 290000 ] ,
               cmap = 'hot_r' , ax = ax[0] ,
               shade = True , shade_lowest = True ,
               cbar = True , clip = usa_cty.geometry
             )
ax[0].annotate( 'Expensive Houses Concentration per km^2', xy = (0.34, 0.06),␣
 ↪xycoords = 'figure fraction' )
ax[0].set_title(" Map 3: Concentration of Houses in California 1990 by top␣
 ↪Quintile ")

#Second HeatMap: Earthquakes concentration

ax [ 1 ] = gplt . polyplot ( usa_cty, ax = ax[1] , zorder = 4 )
gplt.kdeplot( geq, ax = ax[1],
              shade = True, shade_lowest = True,
              cmap= 'hot_r', cbar = True
            )
ax[1].annotate( 'Earthquakes Concentration', xy = (0.86, 0.06), xycoords =␣
 ↪'figure fraction' )
ax[1].set_title( " Map 4: Earthquakes Concentration in California 1969 - 1990 " )

#Saves the map as a png image
plt.savefig("Expensive houses vs. Concentration of Earthquakes.png",␣
 ↪bbox_inches='tight', pad_inches=0.1)
```

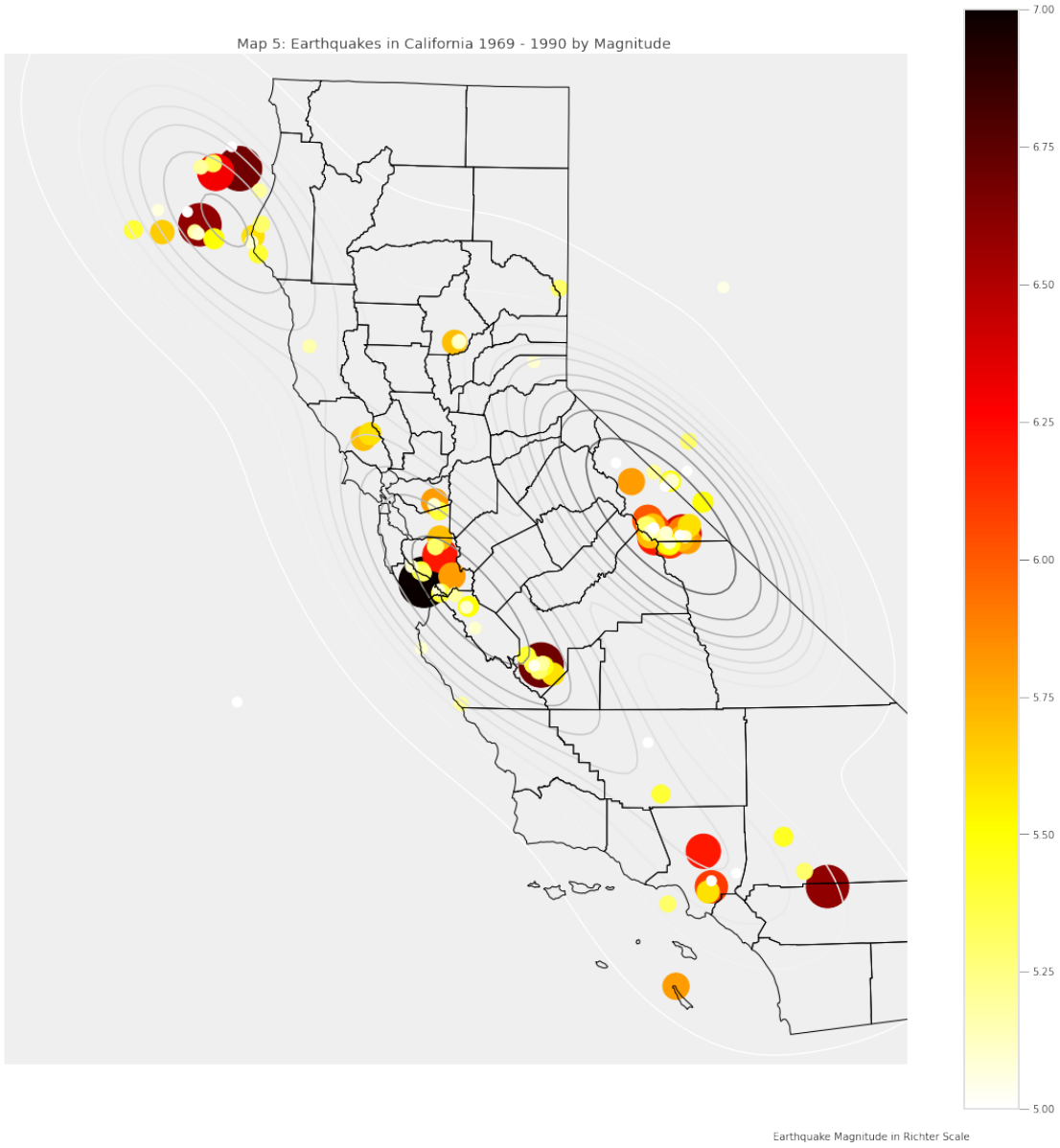




Map 4 shows where the most expensive houses are concentrated in California, from this maps we appreciate a higher concentration near Los Angeles, in the San Francisco bay (norht of the Map) is another concentration of expensive houses but this concentration is lower than the one in Los Angeles. Map 5 shows the concentration of Earthquakes from 1969 to 1990 the interpretation of this map should not be confused with the magnitude of an Eartquake while there is a higher concentration of earthquakes in the nort-east of the map there may be a single earthquake in any area with a higher magnitude. Overall, these two maps show that near the bay area there are a lot of earthquakes in comparision to Los Angeles area, this may be one of the reasons of a higher concentration of expensive properties near Los Angeles. However, is well know that Los Angeles is located over the Fault of San Andreas and the eartquakes follow a random pattern so there is no chance to predict where the next earthquake will take place. Finally, note that those residential buildings located near zones of high earthquakes activity may be expensive on average due to the materials used in their construction (i.e. anti-seismic buildings). To analyze in further detail the magnitude of earthquakes we may need to add another map.

[42]:
```
ax = gplt . polyplot ( usa_cty, figsize = ( 20, 20 ) , projection = gplt.crs.
 ↪AlbersEqualArea( ), zorder = 4 )

gplt.kdeplot( geq, ax = ax,
    shade = False , cmap= 'Greys'
            )

gplt.pointplot( geq,
               ax = ax,
               scale = 'Magnitude', limits = (10 ,50),
               hue = 'Magnitude', cmap = 'hot_r', legend = True
              )

ax.annotate( 'Earthquake Magnitude in Richter Scale' , xy = ( 0.72 , 0.06 ),␣
 ↪xycoords = 'figure fraction' )
ax.set_title("Map 5: Earthquakes in California 1969 - 1990 by Magnitude " )
```

[42]: Text(0.5, 1.0, 'Map 5: Earthquakes in California 1969 - 1990 by Magnitude ')



To sharpen our empirical analysis we add map 6, this map shows the earthquakes in California between 1969 to 1990 by its magnitude. Note that as previous suggested there are three areas of earthquakes concentration while the north- east area concentrates most of the seismic activity the most intense earthquake in this range of years took place near the San Francisco bay area. The north-west location is of our interest since it concentrates three intense earthquakes that my be derived in a tsunami affecting directly to the coast-population (maybe not the best place to purchase a house). Finally, Los Angeles suffered two major earthquakes of around 6 - 6.5 in magnitude but

in comparison with the rest of the state the earthquakes appear to not be common near this area.

---

4 Final Project

4.1 OLS Regression

1) As previously stated our aim is to test the hypothesis that those houses near the ocean tend to be more expensive on average than those that are not so close to the coast. Then, we define the median house price as the dependent variable and the set of dummies together with the set of control variables as the explanatory variables. Thus, we may note that the dummies act like averages for each category, similarly some xs are expected to hold a linear relationship with our dependent variable. As an example consider that those individuals with higher income can access more expensive properties showing a positive linear relationship between these two variables. Also, more bedrooms in a house tend to be a good indicator of the size of the house, therefore more bedrooms higher house prices.

2) The variables that we are going to use are composed of the 8 variables given :'median_house_value', 'median_income', 'housing_median_age', 'total_rooms', 'total_bedrooms', 'population', households'. We run two separate regressions including the variables 'ln_(house_value)', instead of median_house_value and 'ln_(income)', instead of median_income since they contain higher outlier as mentioned in the summary statistics. Also, we consider to include housing_median_age^2' in order to see if the age of a house increases or decreases the value of it at an increasing or decreasing rate. Finally, 'bedroom_rate', and 'household_rate' are included instead of their similars to avoid multicollinearity problems. More detailed explanation can be found in the summary statistics section.

4.1.1 3) Run four regressions and compare estimates

[43]:
```
hc[ 'const' ] = 1
y = ['median_house_value']
y2 = ['ln_(house_value)']

hc['const'] = 1

#naive model
X1 = ['const', 'median_income']

#basic controls
X2 = ['const', 'median_income',
                                        'housing_median_age',
                                        'total_rooms',
                                        'total_bedrooms',
                                        'population',
                                        'households',
```

```
    ␣
 ↪                                                                    '<1H␣
 ↪OCEAN', 'INLAND', 'ISLAND', 'NEAR BAY'

]

#basic controls + ln income
X3 = ['const', 'ln_(income)',
                                        'housing_median_age',
                                        'total_rooms',
                                        'total_bedrooms',
                                        'population',
                                        'households',
    ␣
 ↪                                                                    '<1H␣
 ↪OCEAN', 'INLAND', 'ISLAND', 'NEAR BAY']


#full transformed controls, dependent ln(median_house_value)
X4 = ['const','ln_(income)',
                                        'housing_median_age',
                                        'housing_median_age^2',
                                        'bedroom_rate',
                                        'population',
                                        'household_rate',
                                                                    '<1H␣
 ↪OCEAN', 'INLAND', 'ISLAND', 'NEAR BAY'
]
reg1 = reg1 = sm.OLS(hc[y], hc[X1]).fit()
reg2 = sm.OLS(hc[y], hc[X2]).fit()
reg3 = sm.OLS(hc[y2], hc[X3]).fit()
reg4 = sm.OLS(hc[y2], hc[X4]).fit()
```

[44]:
```
from statsmodels.iolib.summary2 import summary_col

info_dict1 = {'No. observations' : lambda x: f"{int(x.nobs):d}",
                           'Adj R-squared' : lambda x: f"{x.rsquared_adj:2f}",
                         'AIC' : lambda x: f"{x.aic:2f}",
                         'BIC' : lambda x: f"{x.bic:2f}"}

results_table = summary_col(results=[ reg1, reg2, reg3, reg4],
                            float_format='%0.4f',
                            stars = True,
                            model_names=['Naive Model', 'Model 2', 'Model 3',␣
 ↪'Model 4'],
                            info_dict=info_dict1,
                            regressor_order=[ 'const','median_income',
```

```
                                                                        ␣
 ↪'ln_(income)',
                                                                        ␣
 ↪'housing_median_age',
                                                                        ␣
 ↪'housing_median_age^2',
                                                                        ␣
 ↪'total_rooms',
                                                                        ␣
 ↪'total_bedrooms',
                                                                        ␣
 ↪'bedroom_rate',
                                                                         'population',
                                                                         'households',
                                                                        ␣
 ↪'household_rate'])

results_table.add_title(' - OLS Regressions')

results_table
```

[44]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Table 2 - OLS Regressions
============================================================================
                      Naive Model     Model 2       Model 3      Model 4
----------------------------------------------------------------------------
const                 44906.3695*** 39474.0386***  11.1930***   10.4536***
                      (1329.9654)   (2652.4515)    (0.0139)     (0.0263)
median_income         41837.0661*** 40466.1820***
                      (308.4362)    (340.1404)
ln_(income)                                        0.7340***    0.8131***
                                                   (0.0066)     (0.0072)
housing_median_age                  1185.0880***   0.0037***    -0.0021**
                                    (44.3987)      (0.0002)     (0.0008)
housing_median_age^2                                            0.0001***
                                                                (0.0000)
total_rooms                         -7.5095***     -0.0000***
                                    (0.8010)       (0.0000)
total_bedrooms                      81.3102***     0.0004***
                                    (6.9174)       (0.0000)
bedroom_rate                                                    0.0119***
                                                                (0.0006)
population                          -36.9950***    -0.0002***   0.0000***
                                    (1.0880)       (0.0000)     (0.0000)
households                          76.7459***     0.0003***
```

```
                                (7.4822)        (0.0000)
household_rate                                                  0.0123***
                                                                (0.0003)
<1H OCEAN                       -13363.9471***  -0.0276***      -0.0059
                                (1556.1799)     (0.0074)        (0.0072)
INLAND                          -81389.1048***  -0.4987***      -0.4446***
                                (1675.7674)     (0.0080)        (0.0079)
ISLAND                          160788.3925***  0.6625***       0.6048***
                                (31263.0996)    (0.1496)        (0.1437)
NEAR BAY                        -8986.9694***   -0.0238**       -0.0364***
                                (2037.5707)     (0.0097)        (0.0095)
R-squared           0.4738      0.6343          0.6558          0.6824
                    0.4738      0.6345          0.6559          0.6825
No. observations    20433       20433           20433           20433
Adj R-squared       0.473808    0.634311        0.655762        0.682360
AIC                 521221.802496 513795.562626 13179.753875    11536.649901
BIC                 521237.652309 513882.736598 13266.927847    11623.823873
=============================================================================
Standard errors in parentheses.
* p<.1, ** p<.05, ***p<.01
"""
```

4.1.2 4) Selection of models

The four models presented above use the same set of control variables with the difference of slightly transformations between them. Then, the model 1 is a naive model where we only use income as a explanatory variable, the naive analysis suggest that richer people lives in more expensive houses.

However, houses tend to determine their price in the size, location and other factors, therefore in order to control for these factors we add a set of covariates that include aside from the median income, the median age of a house, the population density, the amount of households, the number of rooms, and the number of bedrooms all of these per a given district.

Model 2 uses the set of covariates as controls and we can appreciate that fail to include these controls results in a positive bias for the model 1 since the coefficient for income decreases. As previously stated the transformations to the median price of a house and median income are needed to control for those large outliers that tend to be present in this type of data.

So, we added the model 3 that permits the relation of the natural log of median price of a house and median income and the same set of covariates previously stated.

Finally, model 4 uses a different approach, since population and the number of households share an important correlation factor we suggest a transformation of the variables and instead of using them separately we take the rate of households per 100 inhabitants, this is done to have a rate that can be compared among different sizes of population, that is this is a standarized rate. The same intuition is used to obtain the bedroom rate, which measuares the rate of bedrooms per 100 rooms in each district.

4.1.3 5) - 6) Prefered model

Our prefered model should have three basic characteristics: a high $\bar{R}^2$, and the lowest values of AIC and BIC as possible. Therefore, the selected model is model 4 which collects these criteria. The lowest AIC also know as the Akaike information criterion balances the fit of the model in one side and penalized it for the model complexity on the other side. BIC tends to choose less complex models. Since our dataset is large AIC can be used instead of AICc since the latter converges as n gets large. Adjusted R-squared is the measure when dealing with multivariate regression since it increases only if the coefficients added are significant.

4.1.4 7) Explaining Model 4

The multivariate regression in model 4 has the lowest AIC and BIC with the highest Adj R-squared among the four regressions here analyzed, then we choose this as the best model. This regression suggest that those properties located near the ocean have an average price of $USD\$34,544.37$. Similarly those properties less than one hectare near the ocean have an average price 1 percent lower than our base model (Near Ocean), properties located in land perform even worse with an average price 44 percent points lower and near bay only 4 percent lower that our base. We expect the averages prices of houses located on Island to be 60 percent higher than the average of the houses located near the ocean. Note that all of these estimates are considered while controlling for the other variables. All the variable in dummies are significant at even less than the 1% level except for *<1H OCEAN* which is not significant at any significance level. This is pretty interesting, before including the transformation of model 4, model 3 contains all the coefficients significant at at least the 5% level, however the model 3 shares high levels of multicollinearity among the variables.

4.1.5 Conclusion

Through this paper we have used some statistical techniques in order to investigate any statistically significant relationship between the price of a House in 1990 in California and its location to the ocean. We use an extensive dataset from the California census in 1990 and a set of maps in order to identify any causal relationship. We found, by using maps, that there are observations that are located near the coast but are among the cheapest, we suggest that maybe it is due to their location to centres of intensive earthquake activity. The use of maps in order to analyze the real state market is pretty new and we decide to combine this information with web scrapping techniques in order to extract information about earthquakes. In fact we found, via maps, that houses located near those places that have been heavily struck by earthquakes tend to be cheaper. This type of empirical analysis done by maps is pretty new and must be assessed with statistical relevant information in order to find relevant findings, but still these types of techniques while basic is a good starting point. Thus, using the given dataset we found that those properties located near the ocean tend to be more expensive, on average, to those properties located far away of the coast even at the 1% level.

4.1.6 Future Work

Future research should consider to control for autocorrelation in the error term expanding the basic econometrics models here presented to models that can control for this issue, sharpening the inference tests here presented. Adding more covariates is also recommended since can help to sharpen predictions. Collect data at the district level may be inefficient, as we can see there is

difficult to distinguish exactly between the variables "NEAR OCEAN" and "<1 H OCEAN". In order to avoid this in the future take observations at the county level since it allows to controls for unobservables at the country level. Note that we made a brief analysis of the price of a property and the earthquakes historical in California, future research should consider expand this topic since as previously mentioned those houses near these sites are cheap even if they are near the coast. Measure this type of risk may be difficult once again consider to use fixed effects dummies regression, or go even further with time effects, clustered standard errors at the county level.